AMENDMENTS TO THE CLAIMS

1. (Previously Presented) A computer implemented method for dynamically composing and maintaining applications over a computer architecture comprising:

receiving an indication to dynamically integrate a component into an executing application, wherein the component includes a new component to replace an existing component;

loading the component;

linking the component to the application by obtaining the component's integration interface, the integration interface comprising methods for managing the component, invoking an initialize method of the integration interface; and

invoking a replace method of the integration interface, the replace method to transition an existing state of the existing component into the new component.

2. (Cancelled)

3. (Previously Presented) The method of claim 1, further comprising supporting the component's ability to allow other components to communicate with it by:

invoking a publish method of the integration interface and specifying one or more interfaces to publish to other components; and

storing the one or more interfaces in an interface clearinghouse.

4. (Previously Presented) The method of claim 3, further comprising supporting the component's ability to communicate with other components by:

consulting the interface clearinghouse to determine one or more interfaces to retrieve from other component;

invoking a retrieve method of the integration interface and specifying an interface of the one or more interfaces to retrieve from other component; and

using the retrieved interface to communicate with other components.

5. (Previously Presented) The method of claim 1, further comprising invoking a stop method of the integration interface when the component is ready to be shut down.

6. (Previously Presented) The method of claim 1, wherein the application resides in a network, and the loading of the component comprises retrieving the component from a member in the network.

7. (Original) The method of claim 6, wherein the member comprises a peer.

8. (Original) The method of claim 7, wherein the peer comprises another component loader in the network.

9. (Original) The method of claim 6, wherein the member comprises a host in the network.

10. (Previously Presented) An apparatus comprising:

a client computer system; and

a server computer system coupled with the client computer system, the server computer system including

a component loader to load requested components of a plurality of components into an application, the plurality of components corresponding to an application, and each implementing an integration interface having

a number of methods for managing loaded components,

an initialize method to transition a given component into a state to operate, and a stop method to transition the given component into a destroy state,

an interface clearinghouse to store and manage interfaces corresponding to the loaded components, and

a messaging mechanism for external entities to communicate with the loaded components.

11. (Cancelled)

12. (Previously Presented) The apparatus of claim 10, wherein the server computer system further comprises a replace state to replace an old component with a new component by transitioning an existing state of the old component to the new component.

13. (Previously Presented) The apparatus of claim 10, wherein the application resides in a network, and the loading of the component further comprises retrieving the component from a member in the network.

14. (Previously Presented) A system comprising:

a storage device;

a client computer system coupled with the storage device; and

a server computer system coupled with the client computer system, the server computer system including

an integration interface having a plurality of methods for managing a component,

at least one component that implements the integration interface,

a components repository for storing the at least one component,

a communications bus, wherein the communication bus is established after at least one call to a publish method and a retrieve method of the integration interface, and

a component framework corresponding to an application to manage the at least one component using the integration interface, the component framework having

a component loader to load requested components from the components repository into an application,

an interface clearinghouse to store and manage interfaces corresponding to the loaded components, and

a messaging mechanism for external entities to communicate with the loaded components.

15. (Previously Presented) The system of claim 14, wherein the communication bus is to facilitate inter-component communication.

16. (Cancelled)

17. (Previously Presented) The system of claim 14, wherein the application resides in a network, and the loading of the component comprises retrieving the component from a member in the network.

18. (Currently Amended) A ~~tangible~~ machine-readable medium having stored thereon data representing sets of instructions, the sets of instructions which, when executed by a machine, cause the machine to:

receive an indication to dynamically integrate a component into an executing application, wherein the component is a new component;

load the component;

link the component to the application by

obtaining the component's integration interface, the integration interface comprising methods for managing the component, and

invoking an initialize method of the integration interface; and

replace an existing component by invoking a replace method of the integration interface, the replace method to transition an existing state of the existing component into the new component.

19. (Cancelled)

20. (Currently Amended) The ~~tangible~~ machine-readable medium of claim 18, wherein the sets of instructions which, when executed by the machine, further cause the machine to support the component's ability to allow other components to communicate with it by:

invoking a publish method of the integration interface and specifying one or more interfaces to publish to other components; and

storing the one or more interfaces in an interface clearinghouse.

21. (Currently Amended) The ~~tangible~~ machine-readable medium of claim 20, wherein the sets of instructions which, when executed by the machine, further cause the machine to support the component's ability to communicate with other components by:

consulting the interface clearinghouse to determine one or more interfaces to retrieve from other component;

invoking a retrieve method of the integration interface and specifying an interface of the one or more interfaces to retrieve from other component; and

using the retrieved interface to communicate with other components.

22. (Currently Amended) The ~~tangible~~ machine-readable medium of claim 18, wherein the application resides in a network, and the loading of the component comprises retrieving

the component from a member in the network.

23. (Previously Presented) An apparatus comprising:

at least one processor; and

a machine-readable medium having instructions encoded thereon, which when executed by the processor, are capable of directing the processor to

receive an indication to dynamically integrate a component into an executing application, wherein the component is a new component,

load the component,

link the component to the application by

obtaining the component's integration interface, the integration interface comprising methods for managing the component, and

invoking an initialize method of the integration interface, and

replace an existing component by invoking a replace method of the integration interface, the replace method to transition an existing state of the existing component into the new component.

24. (Cancelled)

25. (Previously Presented) The apparatus of claim 23, the instructions causing the processor to further support the component's ability to allow other components to communicate with it by:

invoking a publish method of the integration interface and specifying one or more interfaces to publish to other components; and

storing the one or more interfaces in an interface clearinghouse.

26. (Previously Presented) The apparatus of claim 25, the instructions causing the processor

to further support the component's ability to communicate with other components by:

consulting the interface clearinghouse to determine one or more interfaces to retrieve from other component;

invoking a retrieve method of the integration interface and specifying an interface of the one or more interfaces to retrieve from other component; and

using the retrieved interface to communicate with other components.

27. (Previously Presented) An apparatus comprising:

means for loading requested components of a plurality of components into an application, the plurality of components corresponding to an application, and each implementing an integration interface having a number of methods for managing loaded components, wherein the integration interface including

means for transitioning a given component into a state to operate, and

means for transitioning the given component into a destroy state;

means for storing and managing interfaces corresponding to the loaded components; and

means for external entities to communicate with the loaded components.

28. (Cancelled)

29. (Previously Presented) The apparatus of claim 27, further comprising means for replacing an old component with a new component by transitioning an existing state of the old component to the new component.

30. (Previously Presented) The apparatus of claim 27, wherein the application resides in a network, and the means for loading the component comprises means for retrieving the component from a member in the network.